

Valeur d'accueil et de reconversion
en informatique (VARI)

(1) Présentation du cours

Daniel Porumbel

dp.cnam@gmail.com

Nombreux slides dus à

- Pierre Cubaud
- Joelle Delacroix
- Emmanuel Hyon
- Bertrand Le cun

Organisation du cours

- 20% architecture, systèmes, réseaux
- 80% programmation

- validation par examen + projet à rendre

- langages utilisés : processing et Java

Site Web:

<http://deptinfo.cnam.fr/new/spip.php?rubrique226>

- 1 Programmes Processing
- 2 Notions d'Architecture

Un premier programme *Processing*

```
int i , j ,somme;  
i=11;  
j=9;  
int somme = i+j;  
println ( "La_somme_vaut_" +somme) ;
```

On observe que les variables sont de type `int`.

Les types principaux sont : `int`, `float`, `char`, `String`, `boolean`

- 1 écrire un programme qui calcule la somme de 3 valeurs de type *float*
- 2 tracer une ellipse
- 3 tracer 4 rectangles collés
- 4 tracer une grille 3×3

Instruction = commande qui définit une action

- la déclaration de variables ;

```
int x
```

- l'affectation, c'est-à-dire donner une valeur (ou le résultat d'une expression) à une variable déclarée,

```
x=9
```

```
x=x+1
```

- la conditionnelle, c'est-à-dire exécuter une suite d'instructions si une expression est vérifiée

```
if (x<10)
```

```
    println(x)
```

Instruction = commande qui définit une action

- la déclaration de variables ;

```
int x
```

- l'affectation, c'est-à-dire donner une valeur (ou le résultat d'une expression) à une variable déclarée,

```
x=9
```

```
x=x+1
```

- la conditionnelle, c'est-à-dire exécuter une suite d'instructions si une expression est vérifiée

```
if (x<10)
```

```
    println(x)
```

```
int i=4;
int j=9;
int somme = i+j;
if (somme<20){
    println("La_somme_vaut_" +somme);
}
```

Le point-virgule

- une instruction de déclaration ou d'affectation sera toujours suivie d'un point-virgule ;
- on dit que le point-virgule valide l'instruction
- nous écrirons une instruction par ligne par souci de clarté.
 - il est possible d'écrire tout le programme en une seule ligne

Le if-else

```
if (a<b){  
    println ("inf");  
} else {  
    println ("sup");  
}
```

à l'intérieur de chaque branche, on peut ajouter d'autres `if`

Le if-else

```
if (a<b) {  
    println ("inf");  
} else {  
    println ("sup");  
}
```

à l'intérieur de chaque branche, on peut ajouter d'autres `if`

```
if (a<b) {  
    println ("inf");  
} else {  
    println ("sup");  
    if (a==b) {  
        println ("=");  
    }  
}
```

Quelques programmes

- écrire un programme qui affiche le maximum de 3 variables a , b et c
- écrire un programme qui affiche “vrai” si 3 entiers a , b et c sont triés $a < b < c$.
- écrire un programme qui trie 3 entiers a , b et c

- 1 Programmes Processing
- 2 Notions d'Architecture**

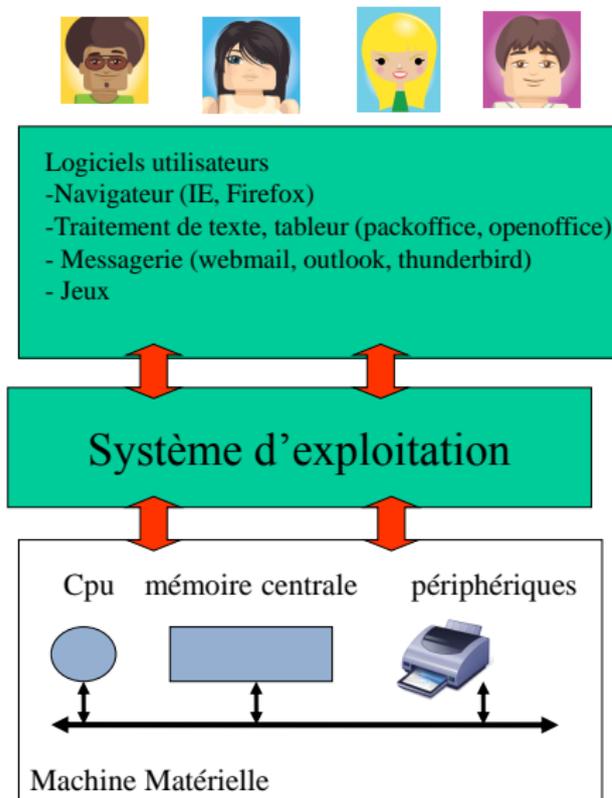
Comprendre l'ordinateur

Pour comprendre le fonctionnement des ordinateurs, on va étudier : [architecture des ordinateurs](#), [systèmes d'exploitation](#), [réseaux](#)

- Le terme "ordinateur" est à prendre dans son sens le plus général, celui d'une "machine électronique capable d'exécuter des opérations arithmétiques et logiques".
- Il peut désigner aussi bien un ordinateur de bureau ou portable (PC, Mac), un serveur de calcul ou encore un terminal mobile de type tablette ou smartphone.

Les différents niveaux de la machine informatique

- On distingue généralement trois couches dans la composition d'une machine informatique :
 - Les **logiciels des utilisateurs « software »** : ce sont des programmes qui permettent à l'utilisateur de réaliser des tâches sur la machine.
 - Le **logiciel de système d'exploitation** : c'est un ensemble de programmes qui se place à l'interface entre le matériel et les logiciels applicatifs. Il permet notamment à ces logiciels applicatifs d'utiliser les ressources matérielles de la machine. Les principaux OS (Operating System) sont notamment Linux, Windows, Mac OS, Unix
 - Le **matériel « hardware »** : il correspond à la machine physique, notamment composée du processeur, de la mémoire centrale et des périphériques, l'ensemble communiquant par un bus.



Les fonctions de l'ordinateur

Les composants

• Les fonctions de l'ordinateur sont de permettre à des utilisateurs (via des logiciels informatiques) de :

- Effectuer du calcul;
- Stocker des données;
- Communiquer.

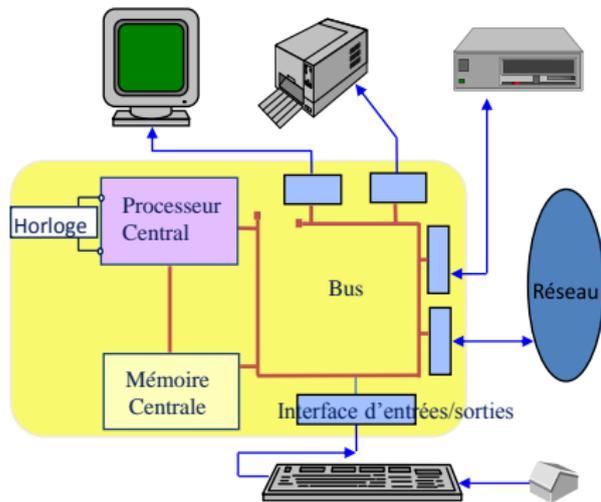
• Pour cela, l'ordinateur est doté d'un ensemble de composants physiques

-Un élément permettant d'exécuter les instructions d'un programme : c'est le **processeur** (CPU).

-Des éléments permettant de stocker les données : ce sont les **mémoires** de l'ordinateur.

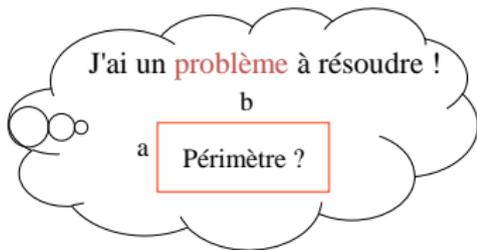
-Des éléments permettant la communication entre l'ordinateur et l'être humain : ce sont les **périphériques**.

-Des éléments permettant aux différents composants (périphériques, processeur, mémoire) de l'ordinateur de communiquer : ce sont les **bus** de l'ordinateur



A quoi sert donc un ordinateur ?

1.



2.



J'écris une solution !
→ ALGORITHME

Périmètre := $2a + 2b$

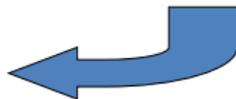


3.

En utilisant un langage de programmation, je code la solution pour la faire exécuter par l'ordinateur

→ PROGRAMME constitué d'instructions

```
fonction perimetre (a, b : in integer) return  
integer is  
begin  
    perimetre := (2 * a) + (2 * b);  
end;
```



Le codage d'un problème ...

```
fonction perimetre (a, b : in integer) return  
integer is  
begin  
    perimetre := (2 * a) + (2 * b);  
end;
```

Programme en langage de haut niveau
instructions de haut niveau



Compilateur

Niveau utilisateur

Système d'exploitation

Gérer et partager le matériel

Machine physique "matérielle"

processeur



Bus

```
01101110111110011  
01111010001011100  
10111101110111111  
00111011110111011  
00111111000111101
```

Mémoire centrale

Programme à
exécuter : instructions machine
et valeurs **en binaire**

traduction

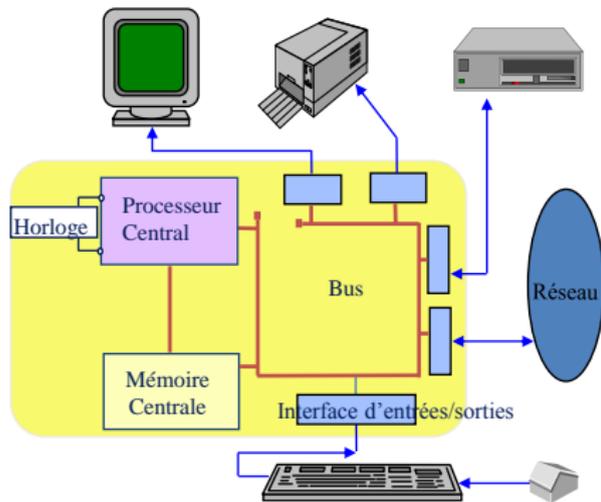
La représentation des informations sur la machine physique

- La donnée de base manipulée par la machine physique est le **bit** (*Binary Digit*) qui ne peut prendre que deux valeurs : 0 et 1
- Ce 0 et 1 correspondent aux deux niveaux de voltage (0-1 et 2-5 volts) admis pour les signaux électriques issus des composants électroniques (transistors) qui constituent les circuits physiques de la machine
- Toutes les informations (nombres, caractères et instructions) ne peuvent être représentées que par une combinaison de 0 et 1 : **chaîne binaire**. Un **octet** est une chaîne de 8 bits.

1100 1011

Les composants de l'ordinateur

- L'ordinateur est doté d'un ensemble de composants physiques
 - Des éléments permettant la communication entre l'ordinateur et l'être humain : ce sont les **périphériques**.
 - Un élément permettant d'exécuter les instructions d'un programme : c'est le **processeur** (CPU).
 - Des éléments permettant de stocker les données : ce sont les **mémoires** de l'ordinateur.
 - Des éléments permettant aux différents composants (périphériques, processeur, mémoire) de l'ordinateur de communiquer : ce sont les **bus** de l'ordinateur

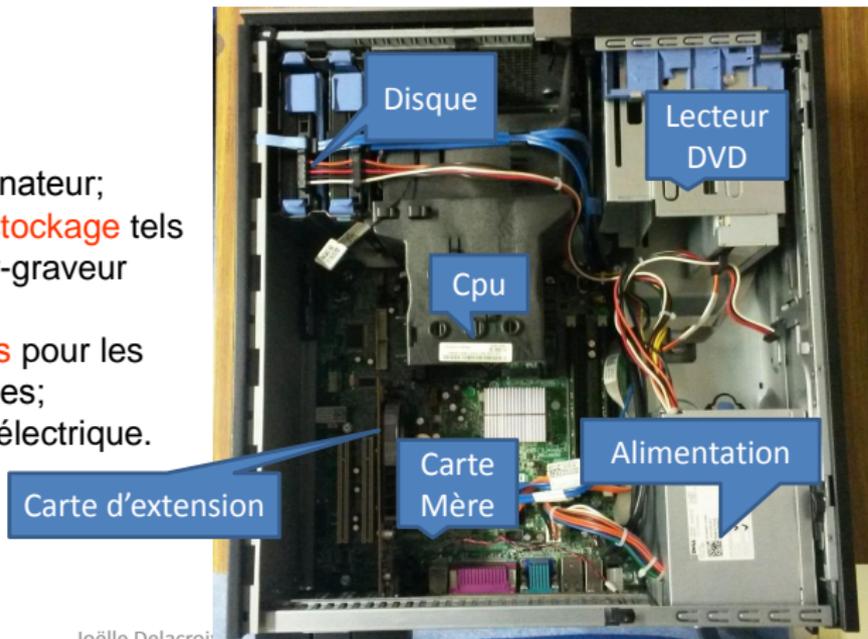


Les composants de l'ordinateur

- Le **boîtier** (ou *châssis*) de l'ordinateur est le squelette métallique abritant ses différents composants internes. L'ensemble, boîtier et composants internes, forment l'**unité centrale** de l'ordinateur.

Le boîtier contient :

- La **carte mère** de l'ordinateur;
- Des **périphériques de stockage** tels que disques, dur, lecteur-graveur DVD.CD-ROM;
- Des **cartes d'extensions** pour les interfaces d'entrées sorties;
- Un **bloc d'alimentation** électrique.

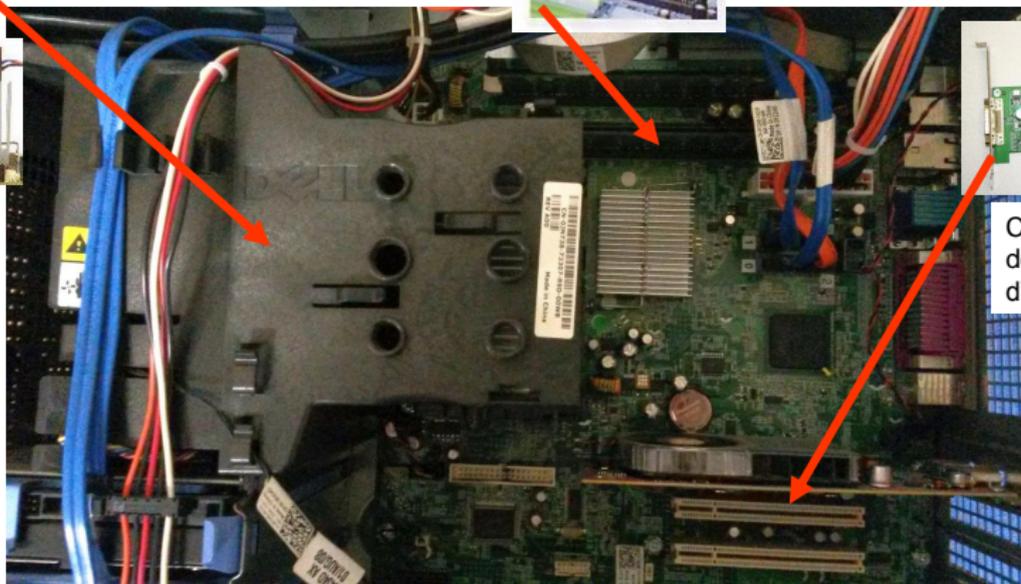


- La **carte mère** de l'ordinateur est le socle permettant la connexion de l'ensemble des éléments essentiels de l'ordinateur.

Processeur et son refroidisseur

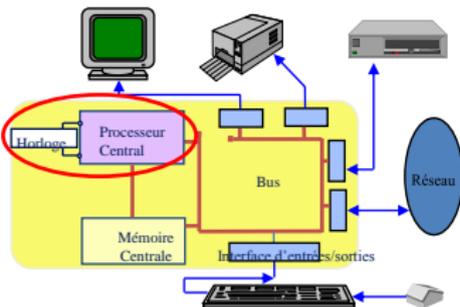


Barrette de mémoire RAM



Connecteurs de cartes d'extensions

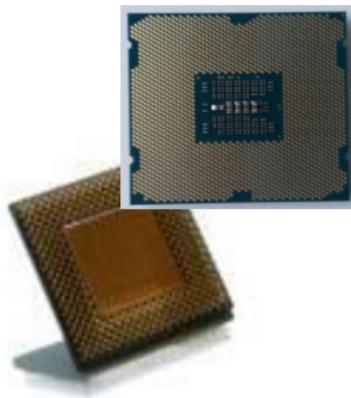
Le processeur



Le **processeur** (CPU, pour *Central Processing Unit*) est le cerveau de l'ordinateur. Il permet de manipuler, des données et des instructions codées sous forme binaires.

Le **processeur** est un circuit électronique cadencé au rythme d'une horloge interne qui envoie des impulsions, appelées « **top** ». La **fréquence d'horloge**, correspond nombre d'impulsions par seconde. Elle s'exprime en Hertz (Hz).

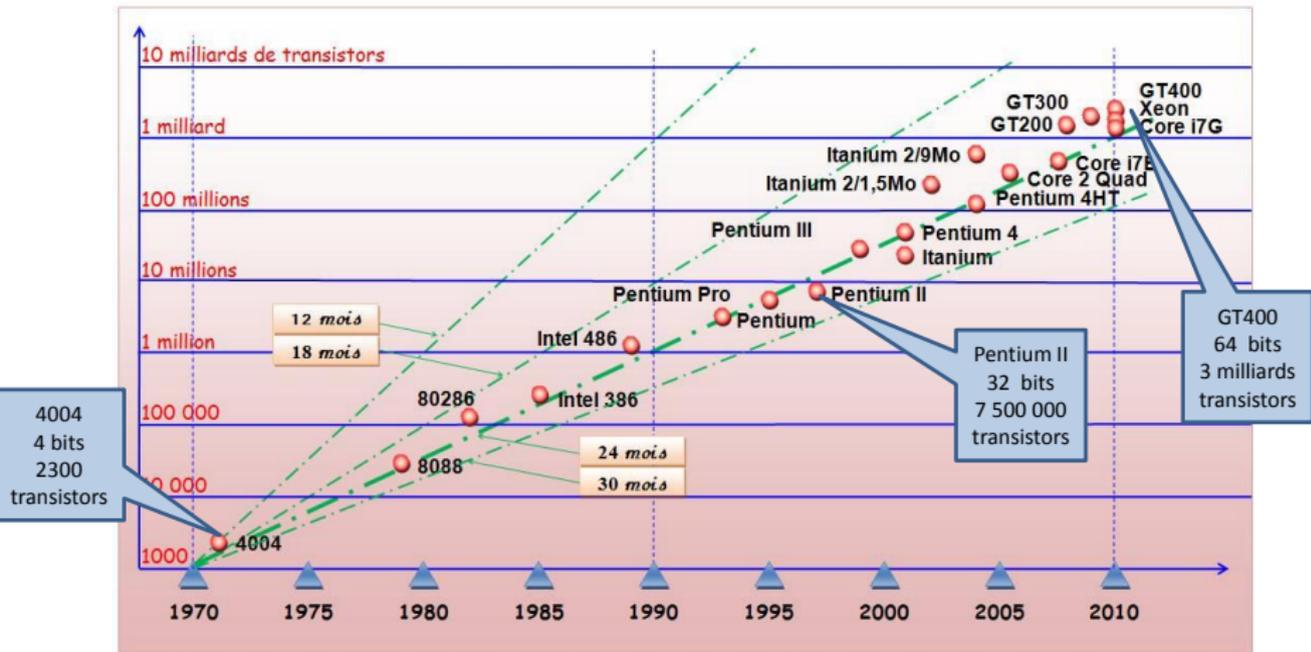
- Ordinateur à 2 GHz → l'horloge envoie 200 000 000 000 battements par seconde.



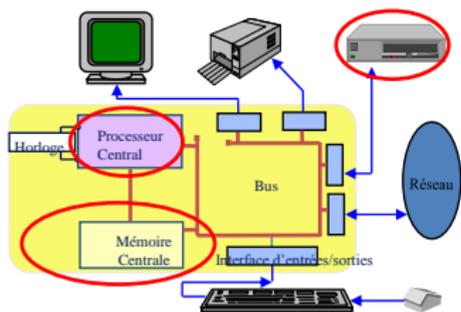
Circuits électroniques composés de millions de transistors placés dans un boîtier comportant des connecteurs d'entrée-sortie, surmonté d'un ventilateur.

→ **circuit intégré** ou **puce**

Le processeur : performances



La **Loi de Moore** a été exprimée en 1965 par *Gordon Moore*, un des trois fondateurs d'Intel. Elle postule que le nombre de transistors sur une puce double tous les deux ans.



Les mémoires de l'ordinateur

Une « **mémoire** » est un composant électronique capable de stocker temporairement des informations

- Une mémoire est caractérisée par :
 - Sa **capacité**, représentant le volume global d'informations (en bits) que la mémoire peut stocker (par exemple 1 Goctets, soit 2^{30} octets, soit $2^{30} * 8$ bits).
 - Son **temps d'accès**, correspondant à l'intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée.
- L'ordinateur contient différents niveaux de mémoire, organisés selon une **hiérarchie mémoire**.

Les grandeurs de l'ordinateur

Capacité – bit - octet

1 octet = 8 bits (byte)	Avant 1998	Après 1998
Kilooctet (Ko)	2^{10} octets = 1024 octets	1000 octets
Mégaoctet (Mo)	2^{20} octets = 1024 Koctets	1000 Koctets
Gigaoctet (Go)	2^{30} octets = 1024 Mcoctets	1000 Mcoctets

Multiples de l'octet :
préfixes SI et mésusages

Nom	Symbole	Valeur	Mésusage ²
kilooctet	ko	10^3	2^{10}
mégaoctet	Mo	10^6	2^{20}
gigaoctet	Go	10^9	2^{30}
téraoctet	To	10^{12}	2^{40}
pétaoctet	Po	10^{15}	
exaoctet	Eo	10^{18}	
zettaoctet	Zo	10^{21}	
yottaoctet	Yo	10^{24}	

Multiples de l'octet :
préfixes binaires

Nom	Symbole	Valeur
kibioctet	kio	2^{10}
mébioctet	Mio	2^{20}
gibioctet	Gio	2^{30}
tébioctet	Tio	2^{40}
pébioctet	Pio	2^{50}
exbioctet	Eio	2^{60}
zébioctet	Zio	2^{70}
yobioctet	Yio	2^{80}

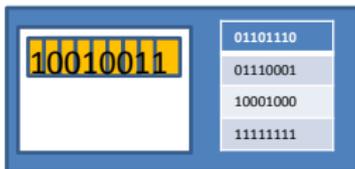
Les mémoires de l'ordinateur

- L'ordinateur contient différents niveaux de mémoire, organisés selon une **hiérarchie mémoire**.

Mémoires vives : mémoires **volatiles** :



Mémoires de masse :
mémoires **permanentes**



REGISTRES
N bits (32, 64)
1 nanoseconde

Mémoires Caches
Koctets (L1,L2)
5 nanosecondes

Mémoires Centrales
Goctets
10 nanosecondes

Mémoires de masse
500 Goctets - Toctets
5 millisecondes

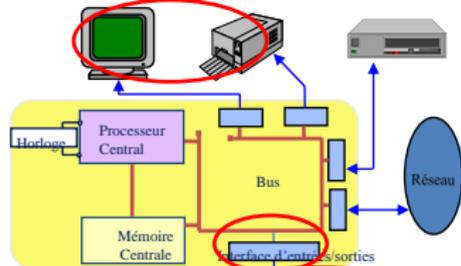
Au plus près du cpu

Capacité, vitesse

Au plus loin du cpu

Mémoire volatile : le contenu de la mémoire n'existe que si il y a une alimentation électrique (typiquement les mémoires caches et mémoire centrale)
Mémoire permanente, de masse : mémoire de grande capacité dont le contenu demeure même sans alimentation électrique (typiquement le disque dur)

Périphériques de l'ordinateur



Un périphérique est un matériel électronique pouvant être raccordé à un ordinateur par l'intermédiaire de l'une de ses **interfaces d'entrée-sortie** (interface VGA, HDMI, USB, RJ45.), le plus souvent par l'intermédiaire d'un **connecteur**. L'interface d'entrées-sorties est pilotée par un **driver (pilote d'entrées-sorties)**



On distingue habituellement les catégories de périphériques suivantes :

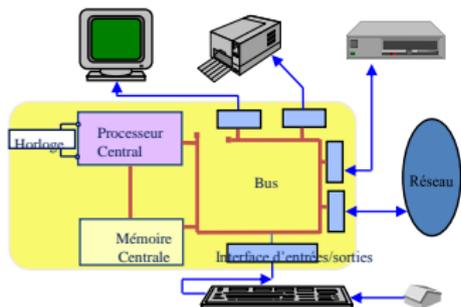
- **périphériques de sortie**: ce sont des périphériques permettant à l'ordinateur d'émettre des informations vers l'extérieur, tels qu'un écran, une imprimante..
- **périphériques d'entrée** : ce sont des périphériques capables uniquement d'envoyer des informations à l'ordinateur, par exemple la souris, le clavier, etc.
- **périphériques d'entrée-sortie** : ce sont des périphériques capables d'envoyer des informations à l'ordinateur et permettant également à l'ordinateur d'émettre des informations vers l'extérieur, par exemple le modem, le disque dur

Interfaces

1. USB : connexion « à chaud » de périphériques
2. RJ45 : connexion au réseau local filaire
3. VGA : connexion de l'écran
4. HDMI : connexion à un écran haute résolution



Les bus de l'ordinateur

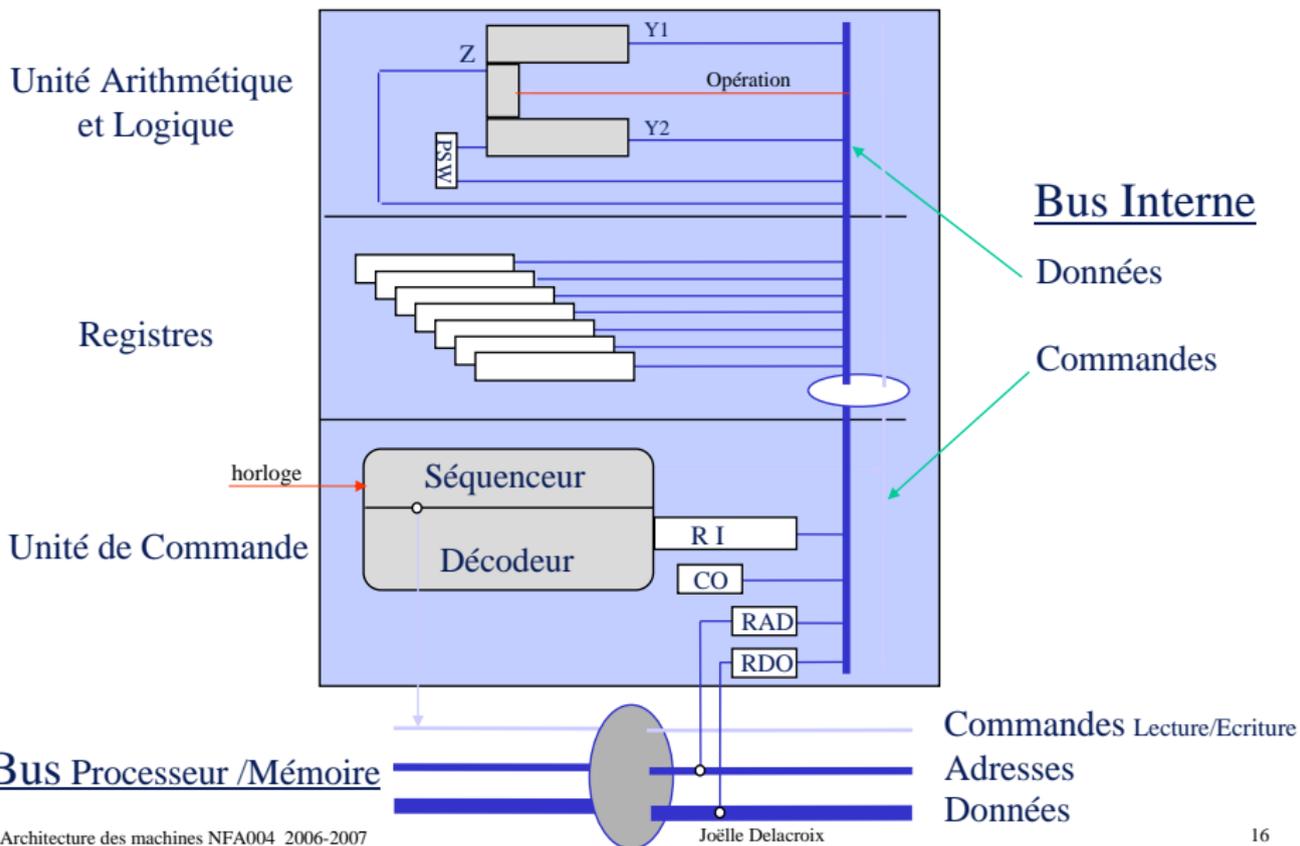


Un « **bus** » est un composant électronique permettant à différents composants de l'ordinateur de s'échanger des informations

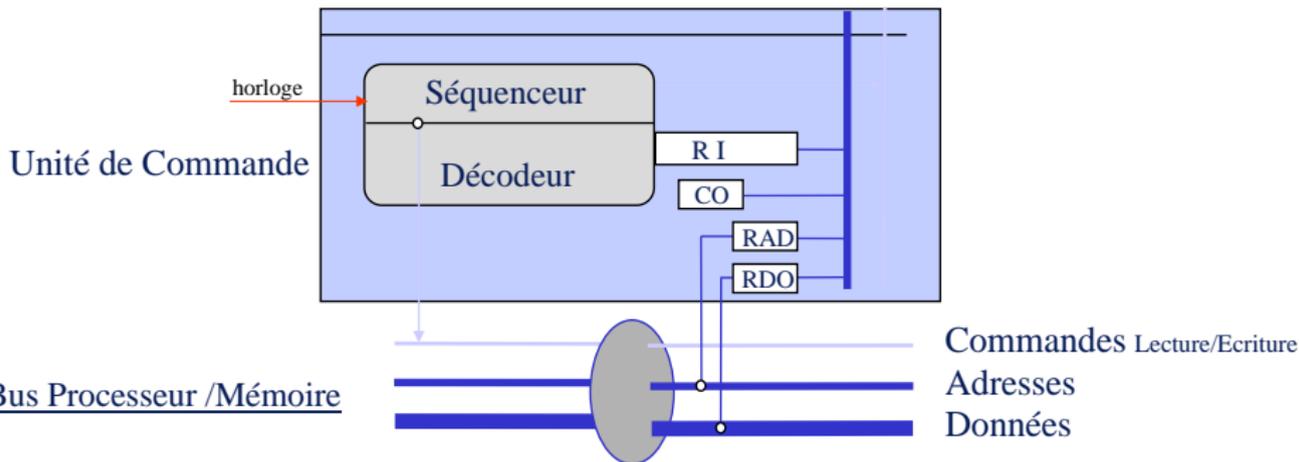
- Bus système (*Front Side Bus FSB*) permet le communication entre le processeur et le mémoire centrale.
- Bus d'extension permet aux autres éléments de l'ordinateur de communiquer entre eux.
- Bus série, bus parallèle, largeur de bus



1. Processeur (Unité Centrale)



1. Processeur (Unité Centrale)

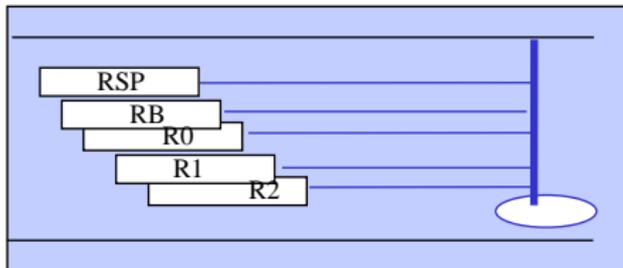


L'unité de commande est chargée de la reconnaissance des instructions et de leur exécution par l'unité de traitement au rythme de l'horloge

Les registres :

- **RI** (registre instruction) : contient l'instruction en cours d'exécution
- **CO** (compteur ordinal) : contient l'adresse en MC de la prochaine instruction
- **RAD** (registre adresse) et **RDO** (registre de données) : registres d'interfaçage avec la mémoire centrale

1. Processeur (Unité Centrale)



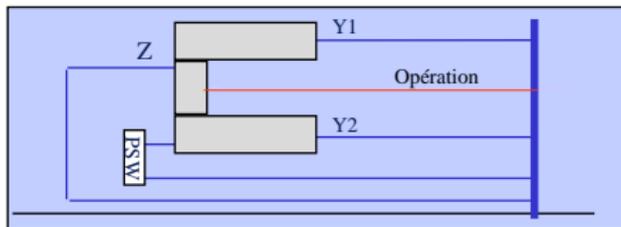
- Le registre est l'entité de base manipulée par le processeur.
- Aucune opération n'est directement réalisée sur les cellules mémoires.

Registres :

- les registres généraux R0, R1, R2
- le registre de pile RSP (Register Stack Pointer)
- les registres d'adressage : RB (registre de base)

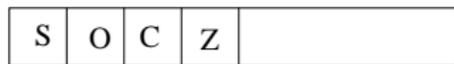
1. Processeur (Unité Centrale)

Unité Arithmétique
et Logique



L'unité Arithmétique et Logique (UAL) constitue l'unité d'exécution du processeur. Elle est composée :

- de l'ensemble des circuits permettant de réaliser les opérations arithmétiques (addition, multiplication, division,...) et les opérations logiques (complément à 2, inverse, OU, ET, ...) sur les opérandes Y1 et Y2
- d'un registre d'état PSW qui contient des indicateurs positionnés par le résultat Z des opérations effectuées :



- O : positionné à 1 si Overflow, 0 sinon
- Z : positionné à 1 si résultat opération nul, 0 sinon
- C : positionné à 1 si carry, 0 sinon
- S : positionné à 0 si résultat opération positif, 1 sinon